CONTRACT DATA REQUIREMENTS LIST
(2 Data Items)

Form Approved
OMB No. 0704-0188

A. CONTRACT LINE ITEM NO.	B. EXHIBIT	C. CATEGORY
0002	A	○ TDP ○ TM ● Other

D. SYSTEM / ITEM	E. CONTRACT / PR NO.	F. CONTRACTOR
Data for CLIN 0001	F25600-02-R-0037	

1. DATA ITEM NO.	2. TITLE OF DATA ITEM	3. SUBTITLE	17. PRICE GROUP
001	Cost Reporting Capability Demo		18. ESTIMATED TOTAL PRICE

4. AUTHORITY *(Data Acquisition Document No.)*	5. CONTRACT REFERENCE	6. REQUIRING OFFICE
	CLIN 0001	55 CONTRACTING SQUADRON - F25600

7. DD 250 REQ	9. DIST STATEMENT REQUIRED	10. FREQUENCY	12. DATE OF FIRST SUBMISSION
LT	D	MTHLY	30DAC
8. APP CODE		**11. AS OF DATE**	**13. DATE OF SUBSEQUENT SUBMISSION**
N/A		0	monthly

16. REMARKS

The contractor shall deliver a capability package demonstrating its ability to report cost activities at the desired levels (e.g., project-level, CLIN level, accounting classification level, etc.). The capability package illustrate the contractor's processes to track cost activities so that decision makers can track costs. Contractor format is acceptable subject to Government approval.

14. DISTRIBUTION	b. COPIES		
a. ADDRESSEE	Draft	Final Reg	Final Repro
SWPS CONTRACT ADMINIS	0	1	0
SWPS PROGRAM CONTROL	0	1	0
15. TOTAL →	0	2	0

1. DATA ITEM NO.	2. TITLE OF DATA ITEM	3. SUBTITLE	17. PRICE GROUP
002	Program Management Plan		18. ESTIMATED TOTAL PRICE

4. AUTHORITY *(Data Acquisition Document No.)*	5. CONTRACT REFERENCE	6. REQUIRING OFFICE
	CLIN 0001	55 CONTRACTING SQUADRON - F25600

7. DD 250 REQ	9. DIST STATEMENT REQUIRED	10. FREQUENCY	12. DATE OF FIRST SUBMISSION
LT	D	ONE/R	1 Nov 02
8. APP CODE		**11. AS OF DATE**	**13. DATE OF SUBSEQUENT SUBMISSION**
N/A		0	19 Dec 02

16. REMARKS

The contractor shall deliver a Program Management Plant. The Program Management Plan will include a conceptual Statement of Work for the development phase and a conceptual Integrated Master Schedule (IMS) for the development phase. Contractor format is acceptabel subject to Government approval.

14. DISTRIBUTION	b. COPIES		
a. ADDRESSEE	Draft	Final Reg	Final Repro
SWPS CONTRACT ADMINIS	0	1	0
SWPS PROGRAM CONTROL	0	1	0
15. TOTAL →	0	2	0

G. PREPARED BY	H. DATE	I. APPROVED BY	J. DATE
CAPT DAVID A. HUBER		Lt. Col. Wynne Waldron	

CONTRACT DATA REQUIREMENTS LIST
(2 Data Items)

Form Approved
OMB No. 0704-0188

A. CONTRACT LINE ITEM NO.	B. EXHIBIT	C. CATEGORY
0002	A	○ TDP ○ TM ● Other

D. SYSTEM / ITEM	E. CONTRACT / PR NO.	F. CONTRACTOR
Data for CLIN 0001	F25600-02-R-0037	

1. DATA ITEM NO.	2. TITLE OF DATA ITEM	3. SUBTITLE	17. PRICE GROUP
003	Risk Management Plan		18. ESTIMATED TOTAL PRICE

4. AUTHORITY *(Data Acquisition Document No.)*	5. CONTRACT REFERENCE	6. REQUIRING OFFICE
	CLIN 0001	55 CONTRACTING SQUADRON - F25600

7. DD 250 REQ	9. DIST STATEMENT REQUIRED	10. FREQUENCY	12. DATE OF FIRST SUBMISSION
LT	D	ONE/R	1 Nov 02
8. APP CODE		**11. AS OF DATE**	**13. DATE OF SUBSEQUENT SUBMISSION**
N/A		0	19 Dec 02

16. REMARKS

The contractor shall deliver a Risk Management Plan. The Risk Management Plan should cite all contractor identified/percieved program risks and how the contractor intends to mitigate those risks. Contractor format is acceptable subject to Government approval.

14. DISTRIBUTION	b. COPIES		
a. ADDRESSEE	Draft	Final Reg	Final Repro
SWPS CONTRACT ADMINIS	0	1	0
SWPS PROGRAM CONTROL	0	1	0
15. TOTAL →	0	2	0

1. DATA ITEM NO.	2. TITLE OF DATA ITEM	3. SUBTITLE	17. PRICE GROUP
004	Software Rqts Specification	(SRS)	18. ESTIMATED TOTAL PRICE

4. AUTHORITY *(Data Acquisition Document No.)*	5. CONTRACT REFERENCE	6. REQUIRING OFFICE
	CLIN 0001	55 CONTRACTING SQUADRON - F25600

7. DD 250 REQ	9. DIST STATEMENT REQUIRED	10. FREQUENCY	12. DATE OF FIRST SUBMISSION
LT	D	ONE/R	1 Nov 02
8. APP CODE		**11. AS OF DATE**	**13. DATE OF SUBSEQUENT SUBMISSION**
N/A		0	19 Dec 02

16. REMARKS

See attached Software Requirements Specification (SRS) instruction pages for content.

14. DISTRIBUTION	b. COPIES		
a. ADDRESSEE	Draft	Final Reg	Final Repro
SWPS CONTRACT ADMINIS	0	1	0
SWPS PROGRAM CONTROL	0	1	0
15. TOTAL →	0	2	0

G. PREPARED BY	H. DATE	I. APPROVED BY	J. DATE
CAPT DAVID A. HUBER		LT. COL. WYNNE WALDRON	

This Software Requirements Specification (SRS) specifies the requirements for a Computer Software Configuration Item (CSCI) A001 The contractor shall develop and deliver for government review and approval an SRS for every system-level CSCI to be developed under the Phase II contract. At a minimum, this shall include the optimization, executive, and decision support functionality, however the software is packaged into one or more CSCIs. Phase I SRSs need not be developed for existing software to be maintained under the Phase II contract (e.g., TIPS, Document Production, Data Services, WAM).This Data Item Description (DID) provides a suggestedformat and content preparation instructions for CSCI A001.

Requirements :

1. Reference documents. None.

2. General instructions.

a. Automated techniques. Use of automated techniques is encouraged. The term "document" in this DID means a collection of data regardless of its medium. If a model, data, or tools-based approach is used, the product(s) shall be capable of being reviewed by the government using STRATCOM-standard office automation tools (e.g., Microsoft Office, standard web browser). At a minimum, artifacts shall be delivered to the government on CD ROM.

b. Alternate presentation styles. Diagrams, tables, matrices, and other presentation styles are acceptable substitutes for text when data required by this DID can be made more readable using these styles.

3. Format. Following are the suggested format. An alternate format is acceptable provided the information contained in the delivered product meets or exceeds the information specified in this DID.. CDRL A001 information shall be delivered on CD-ROM in Microsoft Word format supplemented with other files as appropriate (e.g., files from design tools).

4. Content. The deliverable shall contain the following information:

a. Title page or identifier. The document shall include a title page containing, as applicable: document number; volume number; version/revision indicator; security markings or other restrictions on the handling of the document; date; document title; name, abbreviation, and any other identifier for the system, subsystem, or item to which the document applies; contract number; CDRL item number; organization for which the document has been prepared; name and address of the preparing organization; and distribution statement. For data in a database or other alternative form, this information shall be included on external and internal labels or by equivalent identification methods.

alternative form, files, screens, or other entities shall be assigned names or numbers in such a way that desired data can be indexed and accessed.

d. Response to tailoring instructions. If a paragraph is tailored out of this DID, the resulting document shall contain the corresponding paragraph number and title, followed by "This paragraph has been tailored out." For data in a database or other alternative format, this representation need occur only in the table of contents or equivalent.

e. Multiple paragraphs and subparagraphs. Any section, paragraph, or subparagraph in this DID may be written as multiple paragraphs or subparagraphs to enhance readability.

f. Standard data descriptions. If a data description required by this DID has been published in a standard data element dictionary specified in the contract, reference to an entry in that dictionary is preferred over including the description itself.

g. Substitution of existing documents. Commercial or other existing documents may be substituted for all or part of the document if they contain the required data.

The numbers shown designate the paragraph numbers to be used in the document.

1. Scope. This section shall be divided into the following paragraphs.

1.1 Identification. This paragraph shall contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

1.2 System overview. This paragraph shall briefly state the purpose of the system and the software to which this document applies. It shall describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

1.3 Document overview. This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

2. Referenced documents. This section shall list the number, title, revision, and date of all documents referenced in this document. This section shall also identify the source for all documents.

3. Requirements. This section shall be divided into the following paragraphs to specify the CSCI requirements, that is, those characteristics of the CSCI that are conditions for its acceptance. CSCI requirements are software requirements generated to satisfy the system (TRD) requirements allocated to this CSCI. Each requirement shall be assigned a project-unique identifier to support testing and traceability and shall be stated in such a way that an objective test can be defined for it.

3 .1 Required states and modes. If the CSCI is required to operate in more than one state or mode having requirements distinct from other states or modes, this paragraph shall identify and define each state and mode. Examples of states and modes include idle, ready, active, post-use analysis, training, degraded, emergency, backup, wartime, peacetime. The distinction between states and modes is arbitrary. A CSCI may be described in terms of states only, modes only, states within modes, modes within states, or any other scheme that is useful. If no states or modes are required, this paragraph shall so state, without the need to create artificial distinctions. If states and/or modes are required, each requirement or group of requirements in this specification shall be correlated to the states and modes. The correlation may be indicated by a table or other method in this paragraph, in an appendix referenced from this paragraph, or by annotation of the requirements in the paragraphs where they appear.

3.2 CSCI capability requirements. This paragraph shall be divided into subparagraphs to itemize the requirements associated with each capability of the CSCI. A "capability" is defined as a group of related requirements. The word "capability" may be replaced with "function," "subject," "object, " or other term useful for presenting the requirements.

3.2.x (CSCI capability). This paragraph shall identify a required CSCI capability and shall itemize the requirements associated with the capability .If the capability can be more clearly specified by dividing it into constituent capabilities, the constituent capabilities shall be specified in subparagraphs. The requirements shall specify required behavior of the CSCI and shall include applicable parameters, such as response times, throughput times, other timing constraints, sequencing, accuracy, capacities (how much/how many), priorities, continuous operation requirements, and allowable deviations based on operating conditions. The requirements shall include, as applicable, required behavior under unexpected and "out of bounds" conditions, requirements for error handling, and any provisions to be incorporated into the CSCI to provide continuity of operations in the event of emergencies. Paragraph 3.3.x of this DID provides a list of topics to be considered when specifying requirements regarding inputs the CSCI must accept and outputs it must produce.

3.3 CSCI external interface requirements. This paragraph shall be divided into subparagraphs to specify the requirements, if any, for the CSCI's external interfaces. This paragraph may reference one or more Interface Requirements Specifications (IRSs) or other documents to be developed in Phase II containing these detailed requirements. However, sufficient detail shall be provided in the SRS to allow the government to make an accurate and comprehensive assessment of the overall set of external interface requirements.

3.3 .I Interface identification and diagrams. This paragraph shall identify the required external interfaces of the CSCI (that is, relationships with other entities that involve sharing, providing or exchanging data). The identification of each interface shall include a project-unique identifier and shall designate the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification shall state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams shall be provided to depict the interfaces.

shall include the following, as applicable, presented in any order suited to the requirements, and shall note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):

a. Priority that the CSCI must assign the interface

b. Requirements on the type of interface (such as real-time data transfer, storage-and- retrieval of data, etc.) to be implemented

c. Required characteristics of individual data elements that the CSCI must provide, store, send, access, receive, etc., such as:

1) Names/identifiers

a) Project-unique identifier

b) Non-technical (natural-language) name

c) If applicable, DOD or USSTRATCOM standard data element name

d) Technical name (e.g., variable or field name in code or database)

e) Abbreviation or synonymous names

2) Data type (alphanumeric, integer, etc.)

3) Size and format (such as length and punctuation of a character string)

4) Units of measurement

5) Range or enumeration of possible values (such as 0-99)

6) Accuracy (how correct) and precision (number of significant digits)

7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply

8) Security and privacy constraints

9) Sources (setting/sending entities) and recipients (using/receiving entities)

d. Required characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the CSCI must provide, store, send, access, receive, etc., such as:

c) Technical name {e.g., record or data structure name in code or database)

d) Abbreviations or synonymous names

2) Data elements in the assembly and their structure {number, order, grouping)

3) Medium {such as disk) and structure of data elements/assemblies on the medium

4) Visual and auditory characteristics of displays and other outputs {such as colors, layouts, fonts, icons and other display elements, beeps, lights)

5) Relationships among assemblies, such as sorting/access characteristics

6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply

7) Security and privacy constraints

8) Sources {setting/sending entities) and recipients {using/receiving entities)

e. Required characteristics of communication methods that the CSCI must use for the interface, such as:

1) Project-unique identifier(s)

2) Communication links/bands/frequencies/media and their characteristics

3) Message formatting

4) Flow control {such as sequence numbering and buffer allocation)

5) Data transfer rate, whether periodic/aperiodic, and interval between transfers

6) Routing, addressing, and naming conventions

7) Transmission services, including priority and grade

8) Security considerations, such as encryption, user authentication, compartmentalization, and auditing

f. Required characteristics of protocols the CSCI must use for the interface, such as:

1) Project-unique identifier(s)

5) Synchronization, including connection establishment, maintenance, termination

6) Status, identification, and any other reporting features

g. Other required characteristics, such as physical compatibility of the interfacing entities (e.g., plug compatibility,).

3.4 CSCI internal interface requirements. This paragraph shall specify the requirements, if any, imposed on interfaces internal to the CSCI. If all internal interfaces are left to the design, this fact shall be so stated. If such requirements are to be imposed, paragraph 3.3 of this DID provides a list of topics to be considered.

3.5 CSCI internal data requirements. This paragraph shall specify the requirements, if any, imposed on data internal to the C SCI. Included shall be requirements, if any, on databases and data files to be included in the CSCI. If all decisions about internal data are left to the design, this fact shall be so stated. If such requirements are to be imposed, paragraphs 3.3.x.c and 3.3.x.d of this DID provide a list of topics to be considered.

3 .6 Adaptation requirements. This paragraph shall specify the requirements, if any, concerning installation-dependent data to be provided by the CSCI (such as site-dependent latitude and longitude or site-dependent runtime environment) and operational parameters that the CSCI is required to use that may vary according to operational needs (such as parameters indicating operation-dependent targeting constants or data recording).

3.7 Safety requirements. This paragraph is tailored out.

3.8 Security requirements. This paragraph shall specify the CSCI requirements, if any, concerned with maintaining security. These requirements shall include, as applicable, the security environment in which the CSCI must operate, the type and degree of security to be provided, the security risks the CSCI must withstand, required safeguards to reduce those risks, the security policy that must be met, the security accountability the CSCI must provide, and the criteria that must be met for security accreditation.

3.9 CSCI environment requirements. This paragraph shall specify the requirements, if any, regarding the environment in which the CSCI must operate. Examples include the computer hardware and operating system on which the CSCI must run. (Additional requirements concerning computer resources are given in the next paragraph.) If infrastructure, database, runtime environment, or other environmental changes are required in order for the CSCI to function according to specification, these shall be identified. The contractor is strongly encouraged to pre-coordinate these requirements with the government prior to formal document delivery to ensure that proposed changes are acceptable.

3.10 Computer resource requirements. This paragraph shall be divided into the following subparagraphs

3.10.2 Computer hardware resource utilization requirements. This paragraph shall specify the requirements, if any, on the CSCI's computer hardware resource utilization, such as maximum allowable use of processor capacity, memory capacity, input/output device capacity, auxiliary storage device capacity, and communications/network equipment capacity .The requirements (stated, for example, as percentages of the capacity of each computer hardware resource) shall include the conditions, if any, under which the resource utilization is to be measured to include, at a minimum, average and peak utilization requirements.

3.10.3 Computer software requirements. This paragraph shall specify the requirements, if any, regarding computer software that must be used by, or incorporated into, the CSCI. Example including operating systems, database management systems, communications/network software, utility software, input and equipment simulators, test software, and manufacturing software. The correct nomenclature, version, and documentation references of each such software item shall be provided.

3.10.4 Computer communications requirements. This paragraph shall specify the additional requirements, if any, concerning the computer communications that must be used by the CSCI. Examples include locations to be linked; configuration and network topology; transmission techniques; data transfer rates; gateways; required system use times; type and volume of data to be transmitted/received; time boundaries for transmission/reception/response; peak volumes of data; and diagnostic features.

3.11 Software quality factors. This paragraph shall specify the CSCI requirements, if any, concerned with software quality factors identified in the contract. Examples include quantitative requirements regarding CSCI functionality (the ability to perform all required functions), reliability (the ability to perform with correct, consistent results), maintainability (the ability to be easily corrected), availability (the ability to be accessed and operated when needed), flexibility (the ability to be easily adapted to changing requirements), portability (the ability to be easily modified for a new environment), reusability (the ability to be used in multiple applications), testability (the ability to be easily and thoroughly tested), usability (the ability to be easily learned and used), and other attributes.

3.12 Design and implementation constraints. This paragraph shall specify the requirements, if any, that constrain the design and implementation of the CSCI. These requirements may be specified by reference to appropriate commercial standards and specifications. Examples include requirements concerning:

a. Use of a particular CSCI architecture or requirements on the architecture, such as required databases or other software units; use of standard, military, or existing components; or use of Government/acquirer-furnished property (equipment, information, or software)

b. Use of particular design or implementation standards; use of particular data standards; use of a particular programming language or COTS product

c. Flexibility and expandability that must be provided to support anticipated areas of growth or changes in technology, threat, or mission

particularly serious. Examples include requirements for color and duration of error messages, physical placement of critical indicators or keys, and use of auditory signals.

3.14 Training-related requirements. This paragraph shall specify the CSCI requirements, if any, pertaining to training. Examples include training software to be included in the CSCI.

3.15 Logistics-related requirements. This paragraph shall specify the CSCI requirements, if any, concerned with logistics considerations. These considerations may include: system maintenance, software support, system transportation modes, supply-system requirements, impact on existing facilities, and impact on existing equipment.

3.16 Other requirements. This paragraph shall specify additional CSCI requirements, if any, not covered in the previous paragraphs.

3.17 .Packaging requirements. This section shall specify the requirements, if any, for packaging, labeling, and handling the CSCI for delivery (for example, delivery on CD-Rom labeled and packaging in a certain way). Applicable military specifications and standards may be referenced if appropriate.

3.18 Precedence and criticality of requirements . This paragraph shall specify, if applicable, the order of precedence, criticality, or assigned weights indicating the relative importance of the requirements in this specification. Examples include identifying those requirements deemed critical to security for purposes of singling them out for special treatment. If all requirements have equal weight, this paragraph shall so state.

4. Qualification provisions. This section shall define a set of qualification methods and shall specify for each requirement in Section 3 the method(s) to be used to ensure that the requirement has been met. A table may be used to present this information, or each requirement in Section 3 may be annotated with the method(s) to be used. Qualification methods may include:

a. Demonstration: The operation of the CSCI, or apart of the CSCI that relies on observable functional operation not requiring the use of instrumentation, special test equipment, or subsequent analysis.

b. Test: The operation of the CSCI, or a part of the CSCI, using instrumentation or other special test equipment to collect data for later analysis.

c. Analysis: The processing of accumulated data obtained from other qualification methods. Examples are reduction, interpretation, or extrapolation of test results.

d. Inspection: The visual examination of CSCI code, documentation, etc.

e. Special qualification methods: Any special qualification methods for the CSCI, such as special tools, techniques, procedures, facilities, and acceptance limits.

Note: Each level of system refinement may result in requirements not directly traceable to higher-level requirements. For example, a system architectural design that creates multiple CSCIs may result in requirements about how the CSCIs will interface, even though these interfaces are not covered in system requirements. Such requirements may be traced to a general requirement such as "system implementation" or to the system design decisions that resulted in their generation.

b. Traceability from each system (or subsystem, if applicable) requirement allocated to this CSCI to the CSCI requirements that address it. All system (subsystem) requirements allocated to this CSCI shall be accounted for. Those that trace to CSCI requirements contained in IRSs shall reference those IRSs.

6. Note. This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, or rationale). This section shall include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

Appendices. Appendices may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix shall be referenced in the main body of the document where the data would normally have been provided. Appendixes may be grouped as separate documents for ease in handling..

CONTRACT DATA REQUIREMENTS LIST
(2 Data Items)

Form Approved
OMB No. 0704-0188

A. CONTRACT LINE ITEM NO.	B. EXHIBIT	C. CATEGORY
0002	A	◯ TDP ◯ TM ◯ Other

D. SYSTEM / ITEM	E. CONTRACT / PR NO.	F. CONTRACTOR
Data for CLIN 0001	F25600-02-R-0037	

1. DATA ITEM NO.	2. TITLE OF DATA ITEM	3. SUBTITLE	17. PRICE GROUP
005	Integrated Master Plan (IMP)		18. ESTIMATED TOTAL PRICE

4. AUTHORITY *(Data Acquisition Document No.)*	5. CONTRACT REFERENCE	6. REQUIRING OFFICE
	CLIN 0001	55 CONTRACTING SQUADRON - F25600

7. DD 250 REQ	9. DIST STATEMENT REQUIRED	10. FREQUENCY	12. DATE OF FIRST SUBMISSION
LT	D	ONE/R	1 Nov 02
8. APP CODE		**11. AS OF DATE**	**13. DATE OF SUBSEQUENT SUBMISSION**
N/A		0	19 Dec 02

16. REMARKS

The contractor shall deliver an Integrated Master Plan (IMP). Contractor format is acceptable subject to Government approval.

14. DISTRIBUTION

a. ADDRESSEE	b. COPIES		
	Draft	Final	
		Reg	Repro
SWPS CONTRACT ADMINIS	0	1	0
SWPS PROGRAM CONTROL	0	1	0
15. TOTAL →	0	2	0

1. DATA ITEM NO.	2. TITLE OF DATA ITEM	3. SUBTITLE	17. PRICE GROUP
006	Software Development Plan	(SDP)	18. ESTIMATED TOTAL PRICE

4. AUTHORITY *(Data Acquisition Document No.)*	5. CONTRACT REFERENCE	6. REQUIRING OFFICE
	CLIN 0001	55 CONTRACTING SQUADRON - F25600

7. DD 250 REQ	9. DIST STATEMENT REQUIRED	10. FREQUENCY	12. DATE OF FIRST SUBMISSION
LT	D	ONE/R	1 Nov 02
8. APP CODE		**11. AS OF DATE**	**13. DATE OF SUBSEQUENT SUBMISSION**
N/A		0	19 Dec 02

16. REMARKS

See attached Software Development Plan (SDP) instruction pages for content.

14. DISTRIBUTION

a. ADDRESSEE	b. COPIES		
	Draft	Final	
		Reg	Repro
SWPS CONTRACT ADMINIS	0	1	0
SWPS PROGRAM CONTROL	0	1	0
15. TOTAL →	0	2	0

G. PREPARED BY	H. DATE	I. APPROVED BY	J. DATE
CAPT DAVID A. HUBER		LT. COL. WYNNE WALDRON	

This Software Development Plan (SDP) describes a developer's plans for conducting a software development effort. The term "software development" in this Data Item Description (DID) is meant to include new development, modification, reuse, reengineering, maintenance, and all other activities resulting in software products.

The SDP provides the acquirer insight into, and a tool for monitoring, the processes to be followed for software development, the methods to be used, the approach to be followed for each activity, and project schedules, organization, and resources. This DID contains the format and content preparation instructions for the data product generated.
Portions of this plan may be grouped as separate documents if this approach enhances their usability. Examples include plans for software configuration management and software quality assurance.

Requirements:

1. Reference documents. None.

2. General instructions.

a. Automated techniques. Use of automated techniques is encouraged. The term "document" in this DID means a collection of data regardless of its medium. If a model, data, or tools-based approach is used, the product(s) shall be capable of being reviewed by the government using STRATCOM-standard office automation tools (e.g., Microsoft Office, standard web browser). At a minimum, artifacts shall be delivered to the government on CD ROM.

b. Alternate presentation styles. Diagrams, tables, matrices, and other presentation styles are acceptable substitutes for text when data required by this DID can be made more readable using these styles.

3. Format. Following are the suggested format. An alternate format is acceptable provided the information contained in the delivered product meets or exceeds the information specified in this DID.

4. Content. The plan shall contain the following:

a. Title page or identifier. The document shall include a title page containing, as applicable: document number; volume number; version/revision indicator; security markings or other restrictions on the handling of the document; date; document title; name, abbreviation, and any other identifier for the system, subsystem, or item to which the document applies; contract number; CDRL item number; organization for which the document has been prepared; name and address of the preparing organization; and distribution statement. For data in a database or other alternative format, this information shall be included on external and internal labels or by equivalent identification methods.

b. Table of contents. The document shall contain a table of contents providing the number,

title, and page number of each titled paragraph, figure, table, and appendix. For data in a database or other alternative form, this information shall consist of an internal or external table of contents containing pointers to, or instructions for accessing, each paragraph, figure, table, and appendix or their equivalents.

c. Page numbering/labeling. Each page shall contain a unique page number and display the document number, including version, volume, and date, as applicable. For data in a database or other alternative form, files, screens, or other entities shall be assigned names or numbers in such a way that desired data can be indexed and accessed.

d. Response to tailoring instructions. If a paragraph is tailored out of this DID, the resulting document shall contain the corresponding paragraph number and title, followed by "This paragraph has been tailored out". For data in a database or other alternative form, this representation need occur only in the table of contents or equivalent.

e. Multiple paragraphs and subparagraphs. Any section, paragraph, or subparagraph in this DID may be written as multiple paragraphs or subparagraphs to enhance readability.

f. Standard data descriptions. If a data description required by this DID has been published in a standard data element dictionary specified in the contract, reference to an entry in that dictionary is preferred over including the description itself.

g. Substitution of existing documents. Commercial or other existing documents may be substituted for all or part of the document if they contain the required data

The numbers shown designate the paragraph numbers to be used in the document.

1. Scope. This section shall be divided into the following paragraphs.

1.1 Identification. This paragraph shall contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

1.2 System overview. This paragraph shall briefly state the purpose of the system and the software to which this document applies. It shall describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

1.3 Document overview. This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

1.4 Relationship to other plans. This paragraph shall describe the relationship, if any, of the SDP to other project management plans. This paragraph shall also identify any known changes to the software development process expected during the Phase II effort.

2. Referenced documents. This section shall list the number, title, revision, and date of all documents referenced in this plan. This section shall also identify the source for all documents not available through normal Government stocking activities.

3. Overview of required work. This section shall be divided into paragraphs as needed to establish the context for the planning described in later sections. It shall include, as applicable, an overview of:

a. Requirements and constraints on the system and software to be developed

b. Requirements and constraints on project documentation

c. Position of the project in the system life cycle

d. The selected program/acquisition strategy or any requirements or constraints on it

e. Requirements and constraints on project schedules and resources

f. Other requirements and constraints, such as on project security, privacy, methods, standards, interdependencies in hardware and software development, etc.

4. Plans for performing general software development activities. This section shall be divided into the following paragraphs. Provisions corresponding to non-required activities may be satisfied by the words "Not applicable." If different builds or different software on the project require different planning, these differences shall be noted in the paragraphs. In addition to the content specified below, each paragraph shall identify applicable risks/uncertainties and plans for dealing with them.

4.1 Software; development process. This paragraph shall describe the software development process and methodologies (e.g., object oriented, structured, etc.) to be used. The planning shall cover all contractual clauses concerning this topic, identifying planned builds, if applicable, their objectives, and the software development activities to be performed in each build. This paragraph shall also describe the contractor's approach to spiral development and evolutionary acquisition (EA), including criteria for defining spirals and increments, phases and subphases, entry/exit criteria, planned deliverables, formal and informal reviews, and product correction and quality control processes. Finally, this paragraph shall describe the approach to be used to collaborating with associate contractors (ASCONs) such as other application developers, the government database development and maintenance function, other contractors (e.g., infrastructure providers), and government systems engineers.

4.2 General plans for software development. This paragraph shall be divided into the following subparagraphs.

4.2.1 Software development methods. This paragraph shall describe or reference the

software development methods to be used. Included shall be descriptions of the manual and automated tools and procedures to be used in support of these methods. The methods shall cover all contractual clauses concerning this topic. Reference may be made to other paragraphs in this plan if the methods are better described in context with the activities to which they will be applied.

4.2.2 Standards for software products. This paragraph shall describe or reference the standards to be followed for representing requirements, design, code, test cases, test procedures, and test results. The standards shall cover all contractual clauses concerning this topic. Reference may be made to other paragraphs in this plan if the standards are better described in context with the activities to which they will be applied. Standards for code shall be provided for each programming language to be used. They shall include at a minimum:

a. Standards for format (such as indentation, spacing, capitalization, and order of information)

b. Standards for header comments (requiring, for example, name/identifier of the code; version identification; modification history; purpose; requirements and design decisions implemented; notes on the processing (such as algorithms used, assumptions, constraints, limitations, and side effects); and notes on the data (inputs, outputs, variables, data structures, etc.)

c. Standards for other comments (such as required number and content expectations)

d. Naming conventions for variables, parameters, packages, procedures, files, etc.

e. Restrictions, if any, on the use of programming language constructs or features

f. Restrictions, if any, on the complexity of code aggregates

4.2.3 Reusable software products. This paragraph shall be divided into the following subparagraphs.

4.2.3.1 Incorporating reusable software products. This paragraph shall describe the approach to be followed for identifying, evaluating, and incorporating reusable software products (e.g., COTS, GOTS, newly developed reusable code), including the scope of the search for such products and the criteria to be used for their evaluation and make-vs.-buy decisions. It shall cover all contractual clauses concerning this topic. Candidate or selected reusable software products known at the time this plan is prepared or updated shall be identified and described, together with benefits, drawbacks, and restrictions, as applicable, associated with their use.

4.2.3.2 Developing reusable software products. This paragraph shall describe the approach to be followed for identifying, evaluating, and reporting opportunities for developing reusable software products.

4.2.4 Handling of critical requirements. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for handling requirements designated

critical.

4.2.4.1 Security assurance
4.2.4.3 Assurance of critical requirements

4.2.5 Computer hardware resource utilization. This paragraph shall describe the approach to be followed for allocating computer hardware resources and monitoring their utilization.

4.2.6 Recording rationale. This paragraph shall describe the approach to be followed for recording rationale that will be useful to the support agency for key decisions made on the project. It shall interpret the term "key decisions" for the project and state where the rationale are to be recorded.

4.2.7 Access for acquirer review. This paragraph shall describe the approach to be followed for providing the acquirer or its authorized representative access to developer and subcontractor facilities for review of software products and activities. It shall propose a process which provides the government full and timely insight, review, and approval of all major software development processes and artifacts (e.g., requirements and design specifications; design, test and integration processes; etc.) at all major and incremental program milestones (e.g., requirements baseline establishment, preliminary and final design, test readiness, etc.).

5. Plans for performing detailed software development activities. This section shall be divided into the following paragraphs. Provisions corresponding to non-required activities may be satisfied by the words "Not applicable." If different builds/spirals/increments or different software on the project require different planning, these differences shall be noted in the paragraphs. The discussion of each activity shall include the approach (methods/procedures/tools) to be applied to: 1) the analysis or other technical tasks involved, 2) the recording of results, and 3) the preparation of associated deliverables, if applicable. The discussion shall also identify applicable risks/uncertainties and plans for dealing with them. Reference may be made to 4.2.1 if applicable methods are described there.

5.1 Project planning and oversight. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for project planning and oversight.

5.1.1 Software development planning (covering updates to this plan)
5.1.2 CSCI test planning
5.1.3 System test planning
5.1.4 Software installation planning
5.1.5 Software transition planning
5.1.6 Following and updating plans, including the intervals for management review

5.2 Establishing a software development environment. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for establishing, controlling, and maintaining a software development environment.

5.2.1 Software engineering environment
5.2.2 Software test environment
5.2.3 Software development library
5.2.4 Software development files
5.2.5 Non-deliverable software (Note: It is assumed that all software products and artifacts developed under this contract will be deliverable to the government. If this is not the case (e.g., proprietary solutions), these exceptions shall be noted in this paragraph.)

5.3 System requirements analysis. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for participating in system requirements analysis.

5.3.1 Analysis of user input
5.3.2 Operational concept
5.3.2 Other related system documents and artifacts (e.g., models)
5.3.4 System and software requirements, including any system (e.g., infrastructure) requirements deriving from the contractor's software computing needs

5.4 System design. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for participating in government system design activies. 5.4.1 Overall software architectural design and it's relationship to the system design

5.5 Software requirements analysis. This paragraph shall describe the approach to be followed for software requirements analysis.

5.6 Software design. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for software design.

5.6.1 CSCI-wide design decisions
5.6.2 CSCI architectural design
5.6.3 CSCI detailed design

5.7 Software implementation and unit testing. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for software implementation and unit testing.

5.7.1 Software implementation
5.7.2 Preparing for unit testing
5.7.3 Performing unit testing
5.7.4 Revision and retesting
5.7.5 Analyzing and recording unit test results

5.8 Unit integration and testing. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for unit integration and testing.

5.8.1 Preparing for unit integration and testing
5.8.2 Performing unit integration and testing
5.8.3 Revision and retesting
5.8.4 Analyzing and recording unit integration and test results

5. 9 CSCI qualification testing. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for CSCI qualification testing.

5.9.1 Independence in CSCI qualification testing
5.9.2 Testing on the target computer system
5.9.3 Preparing for CSCI qualification testing
5.9.4 Dry run of CSCI qualification testing
5.9.5 Performing CSCI qualification testing
5.9.6 Revision and retesting
5.9.7 Analyzing and recording CSCI qualification test results

5.10 CSCI/HWCI integration and testing. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for participating in CSCI/HWCI integration and testing.

5.10.1 Preparing for CSCI/HWCI integration and testing
5.10.2 Performing CSCI/HWCI integration and testing
5.10.3 Revision and retesting
5.10.4 Analyzing and recording CSCI/HWCI integration and test results

5.11 System qualification testing. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for participating in system qualification testing.

5.11.1 Independence in system qualification testing
5.11.2 Testing on the target computer system
5.11.3 Preparing for system qualification testing
5.11.4 Dry run of system qualification testing
5.11.5 Performing system qualification testing
5.11.6 Revision and retesting
5.11.7 Analyzing and recording system qualification test results

5.12 Preparing for software use. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for preparing for software use.

5.12.1 Preparing the executable software
5.12.2 Preparing version descriptions for user sites
5.12.3 Preparing user manuals
5.12.4 Installation at user sites (e.g., STRATCOM headquarters, MCCC, NAOC,

TACAMO).

5.13 Preparing for software transition. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for preparing for software transition.

5.13.1 Preparing the executable software
5.13.2 Preparing source files
5.13.3 Preparing version descriptions for the support site
5.13.4 Preparing the "as built" CSCI design and other software support information
5.13.5 Updating the system design description
5.13.6 Preparing support manuals
5.13.7 Transition to the designated support site

5.14 Software configuration management. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for software configuration management. This paragraph shall address managing the multiplicity of software baselines in light of different target environments (e.g., STRATCOM headquarters, MCCC, NAOC, TACAMO).

5.14.1 Configuration identification
5.14.2 Configuration control
5.14.3 Configuration status accounting
5.14.4 Configuration audits
5.14.5 Packaging, storage, handling, and delivery

5.15 Software product evaluation. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for software product evaluation.

5.15.1 In-process and final software product evaluations
5.15.2 Software product evaluation records, including items to be recorded
5.15.3 Independence in software product evaluation, validation, and verification (V&V)

5.16 Software quality assurance. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for software quality assurance.

5.16.1 Software quality assurance evaluations
5.16.2 Software quality assurance records, including items to be recorded
5.16.3 Independence in software quality assurance

5.17 Corrective action. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for corrective action.

5.17.1 Problem/change reports, including items to be recorded (candidate items include project name, originator, problem number, problem name, software element or document affected, origination date, category and priority, description, analyst assigned to the problem, date assigned,

date completed, analysis time, recommended solution, impacts, problem status, approval of solution, follow-up actions, corrector, correction date, version where corrected, correction time, description of solution implemented)

5.17.2 Corrective action system

5 .18 Joint technical and management reviews. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for joint technical and management reviews.

5.18.1 Joint technical reviews (both contractor and government, and contractor, government, and other ASCONs), including a proposed set of reviews
5.18.2 Joint management reviews (both contractor and government, and contractor, government, and other ASCONs), including a proposed set of reviews

5.19 Other software development activities. This paragraph shall be divided into the following subparagraphs to describe the approach to be followed for other software development activities.

5.19.1 Risk management, including known risks and risk-control strategies
5.19.2 Software management indicators, including indicators to be used
5.19.3 Security and privacy
5.19.4 Subcontractor management
5.19.5 Interface with software independent verification and validation (IV& V) agents
5.19.6 Coordination with associate developers
5.19.7 Improvement of project processes
5.19.8 Other activities not covered elsewhere in the plan

6. Schedules and activity network. This section shall present:

a. Schedule(s) identifying the activities in each build and showing initiation of each activity, availability of draft and final deliverables and other milestones, and completion of each activity

b. An activity network, depicting sequential relationships and dependencies among activities and identifying those activities that impose the greatest time restrictions on the project

7. Project Organization and resources. This section shall be divided into the following paragraphs to describe the project organization and resources to be applied in each build.

7.1. Project organization. This paragraph shall describe the organizational structure to be used on the project, including the organizations involved, their relationships to one another, and the authority and responsibility of each organization for carrying out required activities.

7.2 Project resources. This paragraph shall describe the resources to be applied to the

project. It shall include, as applicable:

a. Personnel resources, including:

1) The estimated staff-loading for the project

2) The breakdown of the staff loading numbers by responsibility (for example, management, software engineering, software testing, software configuration management, software product evaluation, software quality assurance)

3) A breakdown of the skill levels, geographic locations, and security clearances of personnel performing each responsibility

b. Overview of developer facilities to be used, including geographic locations in which the work will be performed, facilities to be used, and secure areas and other features of the facilities as applicable to the contracted effort.

c. Acquirer-furnished equipment, software, services, documentation, data, and facilities required for the contracted effort. A schedule detailing when these items will be needed shall also be included.

d. Other required resources, including a plan for obtaining the resources, dates needed, and availability of each resource item.

8. Notes. This section shall contain any general information that aids in understanding this document (e.g., background information. glossary, rationale). This section shall include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terns and definitions needed to understand this document.

A. Appendices. Appendices may be used to provide information published separately for convenience in document maintenance (e.g., charts classified data). As applicable, each appendix shall be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling.

CONTRACT DATA REQUIREMENTS LIST
(2 Data Items)

Form Approved
OMB No. 0704-0188

A. CONTRACT LINE ITEM NO.	B. EXHIBIT	C. CATEGORY
0002	A	○ TDP ○ TM ● Other GDRQ

D. SYSTEM / ITEM	E. CONTRACT / PR NO.	F. CONTRACTOR
Data for CLIN 0001	F25600-02-R-0037	

1. DATA ITEM NO.	2. TITLE OF DATA ITEM	3. SUBTITLE	17. PRICE GROUP
007	System/Subsystem Design Descri	(SSDD)	

4. AUTHORITY *(Data Acquisition Document No.)*	5. CONTRACT REFERENCE	6. REQUIRING OFFICE	18. ESTIMATED TOTAL PRICE
	CLIN 0001	55 CONTRACTING SQUADRON - F25600	

7. DD 250 REQ	9. DIST STATEMENT REQUIRED	10. FREQUENCY	12. DATE OF FIRST SUBMISSION
LT	D	ONE/R	1 Nov 02
8. APP CODE		**11. AS OF DATE**	**13. DATE OF SUBSEQUENT SUBMISSION**
N/A		0	19 Dec 02

16. REMARKS
See attached SSDD instruction page for content.

14. DISTRIBUTION

a. ADDRESSEE	b. COPIES		
	Draft	Final	
		Reg	Repro
SWPS CONTRACT ADMINIS	0	1	0
SWPS PROGRAM CONTROL	0	1	0
15. TOTAL →	0	2	0

1. DATA ITEM NO.	2. TITLE OF DATA ITEM	3. SUBTITLE	17. PRICE GROUP

4. AUTHORITY *(Data Acquisition Document No.)*	5. CONTRACT REFERENCE	6. REQUIRING OFFICE	18. ESTIMATED TOTAL PRICE

7. DD 250 REQ	9. DIST STATEMENT REQUIRED	10. FREQUENCY	12. DATE OF FIRST SUBMISSION
8. APP CODE		**11. AS OF DATE**	**13. DATE OF SUBSEQUENT SUBMISSION**

16. REMARKS

14. DISTRIBUTION

a. ADDRESSEE	b. COPIES		
	Draft	Final	
		Reg	Repro
15. TOTAL →	0	0	0

G. PREPARED BY	H. DATE	I. APPROVED BY	J. DATE
CAPT DAVID A. HUBER		LT. COL. WYNNE WALDRON	

The System/Subsystem Design Description (SSDD) describes the software architecture to be developed by the contractor, to include integration with other applications and system elements (e.g., database, interfaces), COTS and GOTS use, as well as all newly developed Computer Software Configuration Items (CSCIs). The SSDD will be supplemented by Interface Design Descriptions (IDD) and Database Design Descriptions (DBDD) during Phase II; however, the SDD shall be complete and comprehensive enough for the government to evaluate the contractor's proposal without these supplementary documents.

The SSDD, with it associated IDDs and DBDDs, will be used as the basis for further system development. Throughout this Data Item Description (DID), the term "system" may be interpreted to mean "subsystem" as applicable. The resulting document shall be titled System Design Description or Subsystem Design Description (SSDD).

Design pertaining to interfaces may be presented in the SSDD or in IDDs. Design pertaining to databases may be presented in the SSDD or DBDDs.

Requirements:

1. Reference documents. None.

2. General instructions.

a. Automated techniques. Use of automated techniques is encouraged. The term "document" in this DID means a collection of data regardless of its medium. If a model, data, or tools-based approach is used, the product(s) shall be capable of being reviewed by the government using STRATCOM-standard office automation tools (e.g., Microsoft Office, standard web browser). At a minimum, artifacts shall be delivered to the government on CD-ROM.

b. Alternate presentation styles. Diagrams, tables, matrices, and other presentation styles are acceptable substitutes for text when data required by this DID can be made more readable using these styles. An alternate format is acceptable provided the information contained in the delivered product meets or exceeds the information specified in this DID. Whenever possible, the contractor shall use notations consistent with the *DoD Architecture Framework.*

3. Format. The description shall be in contractor format unless otherwise specified. The CDRL shall be delivered in electronic form compatible with Microsoft Word, and may be delivered in developer format rather than in the format specified herein; and may reside in a computer-aided software engineering (CASE) or other automated tool rather than in the form of a traditional word processing document. Following are the suggested format requirements.

a. Title page or identifier. The document shall include a title page containing, as applicable: document number; volume number; version/revision indicator; security markings or

other restrictions on the handling of the document; date; document title; name, abbreviation, and any other identifier for the system, subsystem, or item to which the document applies; contract number; CDRL item number; organization for which the document has been prepared; name and address of the preparing organization; and distribution statement. For data in a database or other alternative form, this information shall be included on external and internal labels or by equivalent identification methods.

b. Table of contents. The document shall contain a table of contents providing the number, title, and page number of each titled paragraph, figure, table, and appendix. For data in a database or other alternative form, this information shall consist of an internal or external table of contents containing pointers to, or instructions for accessing, each paragraph, figure, table, and appendix or their equivalents.

c. Page numbering/labeling. Each page shall contain a unique page number and display the document number, including version, volume, and date, as applicable. For data in a database or other alternative form, files, screens, or other entities shall be assigned names or numbers in such a way that desired data can be indexed and accessed.

d. Response to tailoring instructions. If a paragraph is tailored out of this DID, the resulting document shall contain the corresponding paragraph number and title, followed by "This paragraph has been tailored out." For data in a database or other alternative form, this representation need occur only in the table of contents or equivalent.

e. Multiple paragraphs and subparagraphs. Any section, paragraph, or subparagraph in this DID may be written as multiple paragraphs or subparagraphs to enhance readability.

f. Substitution of existing documents. Commercial or other existing documents may be substituted for all or part of the document if they contain the required data.

4. Content. The numbers shown designate the paragraph numbers to be used in the document.

1. Scope. This section shall be divided into the following paragraphs.

1.1 Identification. This paragraph shall contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

1.2 System overview. This paragraph shall briefly state the purpose of the system and the software to which this document applies. It shall describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

1.3 Document overview. This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

2. Referenced documents. This section shall list the number, title, revision, and date of all documents referenced in this document. This section shall also identify the source for all documents.

3. Software architecture and system-wide design decisions. This section shall be divided into paragraphs as needed to present system-wide software design decisions, that is, decisions about the system's behavioral design (how it will behave, from a user's point of view, in meeting its requirements, ignoring internal implementation) and other decisions affecting the selection and design of system components.

Design decisions that respond to requirements designated critical, such as those for information security, shall be placed in separate subparagraphs. If a design decision depends upon system states or modes, this dependency shall be indicated. Design conventions needed to understand the design shall be presented or referenced. Examples of system-wide design decisions are the following:

a. Design decisions regarding inputs the software will accept and outputs it will produce, including interfaces with other systems, configuration items, and users (4.3.x of this DID identifies topics to be considered in this description). If part or all of this information is to be given in Interface Design Descriptions (IDD), they may be referenced. However, sufficient information shall be provided to enable the government to make an accurate and comprehensive review of the software architecture and related material.

b. Design decisions on system behavior in response to each input or condition, including actions the system will perform, response times and other performance characteristics, description of physical systems modeled, selected equations/algorithms/ rules, and handling of invalid inputs or conditions. Rationale shall be provided supporting each major design decision and algorithm selection, including precedent, prototype, mathematical or engineering analyses, or demonstration.

c. Design decisions on how system databases/data files will appear to the user (4.3.x of this DID identifies topics to be considered in this description). If part or all of this information is given in Database Design Descriptions (DBDDs), they may be referenced.

d. Selected approach to meeting information security, and algorithmic processing (e.g., optimality), and performance requirements.

e. Design decisions regarding openness, software architecture standards used, lifecycle costs, and extensibility to accommodate future requirements, missions, systems, and technologies.

f. Design decisions regarding COTS product selection and make-vs.-buy decisions.

g. Design decisions regarding integration with other system components (e.g., other application software, database, infrastructure and system services, system interfaces).. This shall address the integration approach, technologies (e.g., middleware selections) and products to be

used, and implications for STRATCOM custom and GOTS application reengineering.

h. Proposed infrastructure and database changes, with accompanying rationale. Alternative changes should be provided as appropriate, with cost/benefit/risk tradeoffs clearly articulated. Provide implications if the proposed changes are not approved by the government. (E.g., What is the alternate plan?)

i. Development methodology(ies) selected (e.g., OO, service-based) and accompanying rationale.

j. Design recommendations for hardware systems needed to support the system software design.

k. Other system-wide design decisions made in response to requirements, such as selected approach to providing required flexibility, availability, and maintainability.

4. Softwarearchitectural design. This section shall be divided into the following paragraphs to describe the system software architectural design. If part or all of the design depends upon system states or modes, this dependency shall be indicated. If design information falls into more than one paragraph, it may be presented once and referenced from the other paragraphs. Design conventions needed to understand the design shall be presented or referenced.

Note: For brevity, this section is written in terms of organizing a system directly into Hardware Configuration Items (HWCIs), Computer Software Configuration Items (CSCIs), and manual operations, but should be interpreted to cover organizing a system into subsystems, organizing a subsystem into HWCIs, CSCIs, and manual operations, or other variations as appropriate.

4.1 System components. This paragraph shall:

a. Identify the components of the software system (CSCIs and manual operations). Each component shall be assigned a project-unique identifier and shall include developed software and other system components (e.g., GOTS and other applications).

Note: A software unit is an element in the design of a CSCI. For example, a major subdivision of a CSCI, a component of that subdivision, a class, object, module, function, routine, or database. Software units may occur at different levels of a hierarchy and may consist of other software units. Software units in the design may or may not have a one-to-one relationship with the code and data entities (routines, procedures, databases, data files, etc.) that implement them or with the computer files containing those entities. A database may be treated as a CSCI or as a software unit. The SSDD may refer to software units by any name(s) consistent with the design methodology being used.

b. Show the static (such as "consists of") relationship(s) of the components. Multiple relationships may be presented, depending on the selected design methodology (for example, in

an object-oriented design, this paragraph may present the class and object structures as well as the module and process architectures).

c. State the purpose of each component and identify the system requirements and system-wide design decisions allocated to it. (Alternatively, the allocation of requirements may be provided in 5.a.)

d. Identify each component's development status/type, if known (such as new development, existing component to be reused as is, existing design to be reused as is, existing design or component to be reengineered, component to be developed for reuse, component planned for Build N, COTS, GOTS, etc.) For existing design or components, the description shall provide identifying information, such as name, version, documentation references, location, etc.

e. For each computer system or other aggregate of computer resources identified for use in the system, describe its computer resources (such as processors, memory, input/output devices, auxiliary storage, and communications/network connectivity needed).

1) The requirements being satisfied

2) The assumptions and conditions on which the utilization data are based (for example, typical usage, worst-case usage, assumption of certain events). At a minimum, resource utilization and processing timelines shall be estimated for war planning options containing 1, 10, 20, 50, 100, 1000, 1700, and 2000 targets and weapons.

3) Any special considerations affecting the utilization (such as use of virtual memory, overlays, or multiprocessors or the impacts of operating system overhead, library software, or other implementation overhead)

4) The units of measure used (such as percentage of processor capacity, cycles per second, bytes of memory, kilobytes per second). At a minimum, estimate peak and average processor and memory utilization, processing cycle times (clock time assuming a quiescent environment), disk requirements, and network bandwidth consumption.

5) The level(s) at which the estimates or measures will be made (such as software unit, CSCI, or executable program)

f. Present a specification tree for the system, that is, a diagram that identifies and shows the relationships among the planned specifications for the system components.

4.2 Concept of execution and integration. This paragraph shall describe the concept of execution and integration among the system components. It shall include diagrams and descriptions showing the dynamic relationship of the components, that is, how they will interact during system assembly, storage, deployment, and operation, including, as applicable, flow of execution control, data flow, dynamically controlled sequencing, state transition diagrams,

timing diagrams, priorities among components, handling of interrupts, timing/sequencing relationships, exception handling, concurrent execution, dynamic allocation/deallocation, dynamic creation/deletion of objects, processes, tasks, assembly, storage, deployment, and other aspects of dynamic behavior.

4.3 Integration strategy. This paragraph shall describe in detail the contractor's proposed integration strategy, to include: overall approach (e.g., middleware, database-oriented, etc.); integration components (COTS, developed code, adaptation requirements); implications for other system components (other planning applications, analysis tools, database, network, servers, etc.); expected lifecycle costs and supportability issues; and organizational relationships necessary to implement the approach (e.g., ASCON-to-ASCON agreements). The paragraph shall identify any design issues and tradeoffs.

4.4 Interface design. This paragraph shall be divided into the following subparagraphs to describe the interface characteristics of the system components. It shall include both interfaces among the components and their interfaces with external entities such as other systems, configuration items, and users.

Note: There is no requirement for these interfaces to be completely designed at this level; this paragraph is provided to allow the recording of interface design decisions made as part of system architectural design. If part or all of this information is contained in Interface Design Descriptions (IDD) or elsewhere, these sources may be referenced. However, sufficient information shall be provided to enable the government to make an accurate and comprehensive review of the architecture and related material.

4.4.1 Interface identification and diagrams. This paragraph shall state the project-unique identifier assigned to each interface and shall identify the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification shall state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams shall be provided, as appropriate, to depict the interfaces.

4.4.x (Project-unique identifier of interface) .This paragraph (beginning with 4.3.2) shall identify an interface by project-unique identifier, shall briefly identify the interfacing entities, and shall be divided into subparagraphs as needed to describe the interface characteristics of one or both of the interfacing entities. If a given interfacing entity is not covered by this SSDD (for example, an external system) but its interface characteristics need to be mentioned to describe interfacing entities that are, these characteristics shall be stated as assumptions or as "When [the entity not covered] does this, [the entity that is covered] will...."

This paragraph may reference other documents to be delivered in Phase II (such as data dictionaries, standards for protocols, and standards for user interfaces) in place of stating the information here. The design description shall include the following, as applicable, presented in any order suited to the information to be provided, and shall note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):

a. Priority assigned to the interface by the interfacing entity(ies)

b. Type of interface (such as real-time data transfer, storage-and-retrieval of data, etc.) to be implemented

c. Characteristics of individual data elements that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:

1) Names/identifiers

a) Project-unique identifier

b) Non-technical (natural-language) name

c) DoD or STRATCOM standard data element name

d) Technical name (e.g., variable or field name in code or database)

e) Abbreviation or synonymous names

2) Data type (alphanumeric, integer, etc.)

3) Size and format (such as length and punctuation of a character string)

4) Units of measurement

5) Range or enumeration of possible values (such as 0-99)

6) Accuracy (how correct) and precision (number of significant digits)

7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business roles apply

8) Information Security constraints

9) Sources (setting/sending entities) and recipients (using/receiving entities)

d. Characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the interfacing entity(ies) will provide, store, send, access, receive, etc., such as:

1) Names/identifiers

a) Project-unique identifier

b) Non-technical (natural language) name

c) Technical name (e.g., record or data structure name in code or database)

d) Abbreviations or synonymous names

2) Data elements in the assembly and their structure (number, order, grouping)

3) Medium (such as disk) and structure of data elements/assemblies on the medium

4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights)

5) Relationships among assemblies, such as sorting/access characteristics

6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply

7) Information security constraints

8) Sources (setting/sending entities) and recipients (using/receiving entities)

e. Characteristics of communication methods that the interfacing entity(ies) will use for the interface such as:

1) Project-unique identifier(s)

2) Communication links/bands/frequencies/media and their characteristics

3) Message formatting

4) Flow control (such as sequence numbering and buffer allocation)

5) Data transfer rate, whether periodic/aperiodic, and interval between transfers

6) Routing, addressing, and naming conventions

7) Transmission services, including priority and grade

8) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing

f. Characteristics of protocols that the interfacing entity(ies) will use for the interface, such as:

1) Project-unique identifier(s)

2) Priority/layer of the protocol

3) Packeting, including fragmentation and reassembly, routing, and addressing

4) Legality checks, error control, and recovery procedures

5) Synchronization, including connection establishment, maintenance, termination

6) Status, identification, and any other reporting features

g. Other characteristics, such as physical compatibility of the interfacing entity(ies) (dimensions, tolerances, loads, voltages, plug compatibility, etc.)

6. CSCI design. This section shall be divided into the following paragraphs to describe each software unit of each CSCI to be developed under the contract. If part or all of the design depends upon system states or modes, this dependency shall be indicated. If design information falls into more than one paragraph, it may be presented once and referenced from the other paragraphs. Design conventions needed to understand the design shall be presented or referenced. Interface characteristics of software units may be described here, in Section 4, or in Interface Design Descriptions (IDDs) to be delivered in Phase II. Software units that are databases, or that are used to access or manipulate databases, may be described here or in Database Design Descriptions (DBDDs) to be delivered in Phase II.

6.x (Project-unique identifier of a software unit or designator of a group of software units). This paragraph shall identify a software unit by project-unique identifier and shall describe the unit. The description shall include the following information, as applicable. Alternatively, this paragraph may designate a group of software units and identify and describe the software units and identify and describe the software units in subparagraphs. Software units that contain other software units may reference the descriptions of those units rather than repeating information.

a. Unit design decisions, if any, such as detailed descriptions (e.g., mathematical equations, related data structures, references) algorithms to be used, if not previously selected

b. Any constraints, limitations, or unusual features in the design of the software unit

c. The programming language to be used and rationale for its use if other than the specified CSCI language

d. If the software unit consists of or contains procedural commands (such as menu selections in a database management system (DBMS) for defining forms and reports, on line DBMS queries for database access and manipulation, input to a graphical user interface (GUI) builder for automated code generation, commands to the operating system, or shell scripts), a list of the procedural commands and reference to user manuals or other documents that explain them.

e. If the software unit contains, receives, or outputs data, a description of its inputs, outputs, and other data elements and data element assemblies, as applicable. Paragraph 4.3.x of this DID provides a list of topics to be covered, as applicable. Data local to the software unit shall be described separately from data input to or output from the software unit. If the software unit is a database, a corresponding Database Design Description (DBDD) to be delivered during Phase II shall be referenced; interface characteristics may be provided here or by referencing section 4 or the corresponding Interface Design Description(s).

f. If the software unit contains logic, the logic to be used by the software unit, including, as applicable:

1) Conditions in effect within the software unit when its execution is initiated

2) Conditions under which control is passed to other software units

3) Response and response time to each input, including data conversion, renaming, and data transfer operations

4) Sequence of operations and dynamically controlled sequencing during the software unit's operation, including:

a) The method for sequence control

b) The logic and input conditions of that method, such as timing variations, priority assignments

c) Data transfer in and out of memory

d) The sensing of discrete input signals, and timing relationships between interrupt operations within the software unit

5) Exception and error handling

6 .Requirements traceability .This section shall contain:

a. Traceability from each system component identified in this SSDD to the system requirements (SRS and TRD) allocated to it. (Alternatively, this traceability may be provided in 4.1.)

b. Traceability from each system requirement to the system software component(s) to which it is allocated.

7. Notes. This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall include an

alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

A. <u>Appendices.</u> Appendices may be used to provide information published separately for convenience in document maintenance (e.g., charts classified data). As applicable, each appendix shall be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling.